

# MOBILE ACCESS TO THE SERVICES IN AMBIENT NETWORKS

A. Reyes\*, R. Messeguer\*, D. Royo\*, C. Homar\*

\*Technical University of Catalonia, Spain, messeguer@ac.upc.edu

**Keywords:** Ambient Networks, Ambient Intelligence, Mobile Access, Web Services.

## Abstract

Ambient Networks will enable mobile terminals to communicate in an environments populated by several networks, devices, services and artefacts in a transparent way to the final user. In these scenarios, one way of implementing services uses web service architecture. So it is important to study how these mobile devices will integrate efficiently to access web services. Because of the reduced capabilities of the mobile devices, their access to web services is not easy and might have some performance costs related to show data speeds and processing both HTTP commands and XML. Moreover, some mobile devices could not support web service access and if supported it lacks of web service search engine. In order to allow mobile devices to access web services efficiently, giving a solution to the previous problems, we propose a new architecture based on a peer-to-peer with distributed hash tables network, as a gateway between the mobile device and web service. This architecture is a base software simulator for an Ambient Network environment developed in our research group. Nodes of the peer-to-peer network publish services, allow finding them easily and acting as a gateway to the mobile devices for accessing web services more efficiently.

## 1 Introduction

Our research has been focused on the prototyping of Ambient Networks (AN) [1] based on Peer to Peer (P2P) Distributed Hash Table (DHT) architectures that provides services implemented using web service architecture [6].

Currently, several kind of mobile devices are indispensable tools in our daily life, so using mobile device for accessing services has much significance, because people that own the device can access to the service anytime, anywhere.

In our prototype, services are implemented using web service architecture. For desktop computer, there are near mature software develop SDK and models to build, or access web services. However, things are not the same for

mobile devices, due to resource constraints for mobile devices such as memory and bandwidth.

One solution to this problem is to build a gateway that sits between the mobile device and the web service to handle these problems providing an efficient way to web service access. In [2] [5] [8] several strategies for deploying a gateway are described, in all of them it is shown that using the gateway improves web service response time, but none of them supports the web service finding engine.

In this paper, we propose P2P DHT- based architecture, which allows mobile devices to consume but also to find web services efficiently.

## 2 Ambient Networks Scenario

In the last years, all technological development is following the idea of creating full-connected environments, where users will be able to access many services and devices in a completely transparent way.

Ambient Networks (ANs) look for a mobile network design between heterogeneous networks leaving the particular characteristics of each network in a subjacent level.

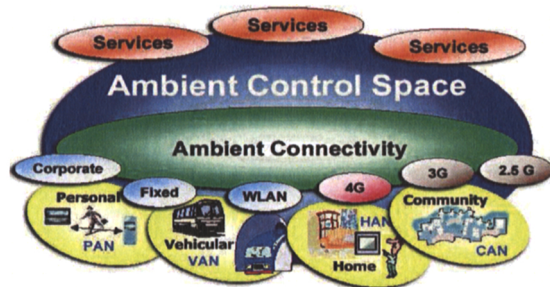


Figure 1: Ambient Network environment.

Basically, there are four methods to simulate an AN scenario [4][7]: Pattern-based approach, P2P approach, Plug and play approach and Closed-loop- based approach (also known as Traffic Engineering)

We have modelled an AN as a P2P network based on FreePastry [6]. So, the P2P paradigms are used for the maintenance of the hierarchy, as well as for solving management and control tasks.

Besides this, our network scenario benefits from some Pastry characteristics such as: efficient request routing,

deterministic object location and load balancing in an application-independent manner.

In this way, P2P nodes publish services and the DHT facilitates the work of finding the published services in our ambient network environment.

Services in our architecture are implemented through a web service scheme, which provides high interoperability due to all the interfaces are independent from hardware and software.

### 3 Mobile-Device Access Web Service Architecture

The web services architecture has three distinct roles (See fig. 2a) [5]: a provider, a requestor and a registry. The provider creates the web service and makes it available to clients who want to use it. A requestor is a client application that consumes the web service. The registry, such a service registry, provides a way for the provider and the requestor of a web service to interact. The requestor queries the registry to find services and the provider sends the web service description to the registry for publishing web services.

In a basic service-oriented architecture for Java mobile-devices (See fig 2.b), the requestor is the mobile-device. In this scenario, there are to main problems: standard J2ME for mobile-devices (JSR172) [3] does not provide discovery services for mobile devices and because hardware and software limitations of mobile-devices consuming web services are very inefficient.

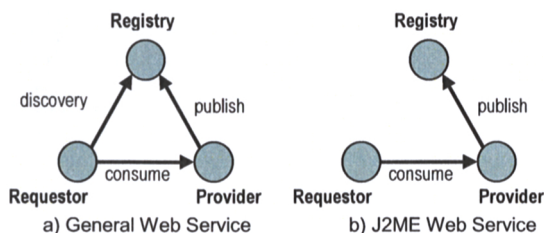


Figure 2. Web Service Architecture (fixed and mobile)

To solve these problems we propose the architecture described in figure 3. In this architecture our Ambient Network prototype based on Pastry P2P DHT (SNAF [6]) performs registry role. All nodes of the prototype have capabilities for publishing, discovery and consume web services.

To efficiently integrate our prototype with service-oriented architecture for Java mobile-devices one P2P DHT node, the gateway, interacts with mobile-devices. Through this gateway mobile-devices are capable of querying the registry (discovery) to get web service information.

At this point, mobile-devices have enough information about web services for consuming them directly.

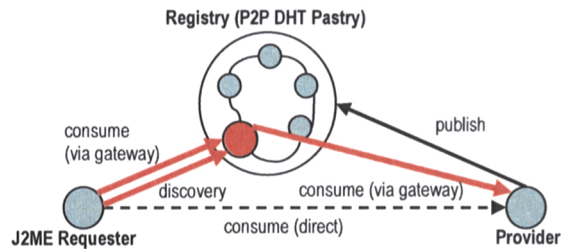


Figure 3. Architecture to consume-discover Web Services from mobile devices

As we have pointed before, because of the reduced capabilities of the mobile devices, their access to web services is not easy and might have some performance costs related to show data speeds and processing both HTTP commands and XML. That is why, in our architecture, mobile-devices have the option of consuming web services through the gateway.

In next section, some experiments show that in mobile scenario, consuming web services via the gateway is faster than consuming directly from mobile-devices.

### 4 Experiment on accessing Web Service from Mobile-Device

In this section we describe experiments for testing our proposal. In order to gain a better understanding of the issues associated with web services consumption of our architecture, we configured a mobile device Nokia N80 Internet Edition to test the application. The device had WLAN connectivity and supports Java MIDP 2.0 and CLDC 1.1.

Three different clients, direct, proxy and fixed, have been implemented for testing our architecture:

1. Proxy. The gateway (MIDlet ProxyWS) placed in a node of P2P network, consumes the service. The mobile device sends the parameters to the gateway and receives the response from the gateway.
2. Direct. Mobile-device consumes directly web services (MIDlet DirectWS).
3. Fixed. A P2P DHT node consumes the web service.

In all tests, our prototype SNAF [6] is in charge of publishing and discovering web services.

We have implemented two web services: CalculadoraWS and HolaMundoWS.

The characteristics we have evaluated are:

- a) Number of bytes exchanged between web service and the client.
- b) Time to invoke a web service method and time to receive the response.

- c) Number and type of messages exchanged between client and web services.

#### a) Number of bytes

Figure 4 shows the number of exchanged bytes when clients consume web service method.

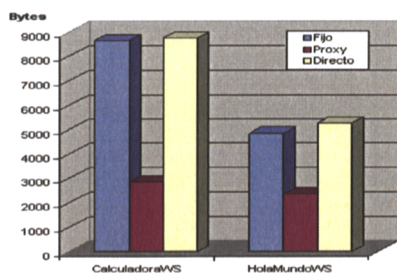


Figure 4: Number of transmitted Bytes for each service.

- Fixed and direct clients exchange a very similar number of bytes because both of them do the same operations.
- The number of exchanged bytes depends on the web service, the more are the functions of the web service the bigger is its WSDL file.
- When we use a proxy client, it has to unload the WSDL file and to communicate with the web service. So the load of the Pastry network is increased.

#### b) Time to invoke/receive a method

Figure 5 shows the time to invoke a web service method and the required time to receive web service response.

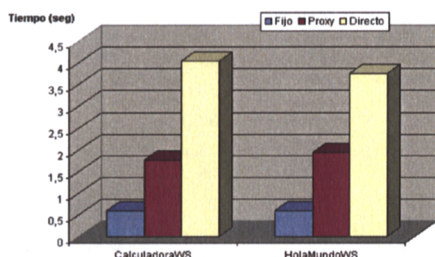


Figure 5: Web services execution time.

- Execution time from fixed clients is lower than time from mobile clients due to several reasons, for example the throughput of wireless connection is lower than the throughput of fixed systems.
- The gateway client needs less time than direct client. The reason is that direct client decodifies the WSDL file, while in case of gateway clients the pastry node decodifies the WSDL file.
- Decoding time increases according with the number of functions that have the web service, so the execution time also is increased.

#### c) Messages exchanged between client and service

Next table shows the number of exchanged messages using two different web services. As we can see there is no significant variation between them.

	Cliente Fijo	Cliente Proxy	Cliente Directo
CalculadoraWS	24	29	23
HolaMundoWS	20	24	20

Table 1: Number of exchanged messages.

- Direct client starts a TCP connection with the web service and later exchanged SOAP messages over HTTP.
- Gateway client only exchanges its own TCP messages of the socket-proxy connection.

Performance tests of different web services in high demand produced similar results in execution time.

With these results, we conclude that mobile web service consumption using our architecture is much faster than using J2ME.

## 5. Conclusions and future work

In this paper, we contribute to increase the potential for web services on mobile devices. There are several challenges on this area due to mobile devices have much lower processor power, limited bandwidth and less memory than desktop computers. Besides this, web services specifications do not consider some mobility issues such as service discovery. Therefore, we have implemented a gateway-based architecture that distributes the web services along network nodes which performs publish, discovery and consumption web service roles.

Currently, web service consumption on mobile devices basically can be done in two ways: mobile device consumes the service directly by itself or mobile device through a web service gateway technology, which makes consume more efficient.

Finally, we have described how service search and consumption functions are made through out our proposed gateway and have proved that our architecture helps to overcome the lack of standard J2SE specifications to discover web services from mobile devices as well as device limitations.

## Acknowledgements

This research partially supported by the Spanish MEC project P2PGrid TIN2007-68050-C03-01.

## References

- [1] Ambient Networks <http://www.ambient-networks.org/>
- [2] S. Berger, S. McFaddin, C.Narayanaswami, M.Raghunath. Web Services on Mobile Devices – Implementation and Experience *Proceedings of the Fifth IEEE Workshop on Mobile Computing Systems & Applications*. 2003
- [3] J2ME Specification for Web Services <http://developers.sun.com/mobility/apis/articles/wsa/>

- [4] P. Kersch, R. Szabo, Z. Lajos, Self Organizing Ambient Control Space. An Ambient Network Architecture for Dynamic Network Interconnection, *International Conference on Mobile Computing and Networking. Dynamic Interconnection of Networks*. 2005
- [5] Luqun Li, Minglu Li, Xianguo Cui. The Study on Mobile Phone-Oriented Application Integration Technology of Web Services. *Second International Workshop of Grid and Cooperative Computing*, pp. 867-874, Shanghai China, December 2003.
- [6] D. Royo, A. Reyes, R. Messeguer, R. Martínez. SNAF: A Service Naming and Addressing Framework for SerAN Platform, *3rd IET International Conference on Intelligent Environments*. Ulm Germany 2007.
- [7] A. Schieder, L. Eggert, N. Papadoglou, F. Pittmann, Components and Concepts of the Ambient Networks Architecture. *Wireless World Research Forum (WWRF)*. 2004
- [8] Yoshiro Imai, Yuichi Sugiue, Yukio Hori, Yutaka Iwamoto, Shin'ichi Masuda: An Enhanced Application Gateway for some Web services to Personal Mobile Systems, *International Conference on Computational Intelligence for Modelling Control and Automation*, pp. 1055-1060, Vienna Austria, November 2005.